e-ISSN 2962-6129
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 20 Agustus 2022

KLASIFIKASI SENTIMEN DATA TIDAK SEIMBANG MENGGUNAKAN ALGORITMA SMOTE DAN *K-NEAREST NEIGHBOR* PADA ULASAN PENGGUNA APLIKASI PEDULILINDUNGI

Sheila Gabriela Barus
Prodi S1 Informatika / Fakultas Ilmu Komputer
Universitas Pembangunan Nasional Veteran Jakarta
Jl. RS. Fatmawati Raya, Pd. Labu, Kec. Cilandak, Kota Depok, Daerah Khusus Ibukota Jakarta 12450
sheilagb@upnvj.ac.id

Abstrak. Salah satu penanganan pemerintah dalam mengatasi penyebaran Covid-19 yang terjadi di Indonesia yaitu dengan membuat sebuah aplikasi yaitu aplikasi PeduliLindungi. Aplikasi ini berfungsi dalam melacak dan memantau penyebaran Covid-19, oleh karena itu banyak masyarakat Indonesia yang harus mempunyai aplikasi ini. Banyak juga ulasan yang diberikan pada aplikasi ini, dari komentar yang positif hingga komentar negatif. Ulasan tersebut yang menjadi data dalam penelitian ini untuk mengetahui hasil sentimen masyarakat dan menguji klasifikasi algoritma *K-Nearest Neighbor*. Pengumpulan data dilakukan dengan *scraping* di google play menggunakan bahasa pemrograman *Python*, dimana data yang diperoleh mendapatkan 750 label negatif dan 250 label positif. Sehingga data yang tidak seimbang ini harus diseimbangkan dengan teknik *undersampling* dan *oversampling* SMOTE. Oleh karena itu, penelitian ini dilakukan tiga data yang berbeda jumlah yaitu dari data yang tidak seimbang, data yang sudah di *undersampling* dan data yang sudah di *oversampling* dengan SMOTE. Hasil dari ketiga percobaan tersebut diperoleh nilai terbaik menggunakan teknik SMOTE pada K = 1 dengan nilai akurasi sebesar 0.9766, nilai presisi sebesar 0.9691, nilai *F1 score* 0.9781, nilai spesifisitas sebesar 0.9645, dan nilai sensitivitas sebesar 0.9874.
Kata Kunci : Sentimen, *K-Nearest Neighbor*, SMOTE, PeduliLindungi

1 Pendahuluan

Awal tahun 2020, dunia dihebohkan dengan mewabahnya virus baru yaitu virus corona baru (SARSCoV2) yang bernama coronavirus disease 2019 (Covid-19). Virus ini diketahui terjadi di Wuhan, China. Ditemukan pada akhir Desember 2019. Sejauh ini, 29 negara telah dipastikan terinfeksi virus ini (Data WHO, 1 Maret 2020) (PDPI, 2020). Covid-19 pertama dilaporkan dalam dua kasus di Indonesia pada 2 Maret 2020. Menurut data per 31 Maret 2020, 1.528 kasus terkonfirmasi dan 136 kematian terkonfirmasi. Angka kematian Covid-19 di Indonesia adalah 8,9%, tertinggi di Asia Tenggara (Susilo et al., 2020). Meningkatnya pandemi Covid-19 di beberapa negara telah menciptakan krisis global baik di sektor ekonomi maupun kemanusiaan. Krisis kemanusiaan terjadi ketika jumlah infeksi dan kematian di seluruh dunia mencapai jutaan dan kemungkinan akan terus meningkat karena pandemi belum ditentukan kapan pandemi akan berakhir (Nurhidayati et al., 2021). Salah satu terobosan pemerintah di Indonesia dalam penanganan Covid-19 adalah pengembangan aplikasi PeduliLindungi. Aplikasi ini dirancang untuk mengingatkan masyarakat umum saat memasuki area terdampak Covid-19, lokasi fasilitas medis, dan melacak orang yang mungkin terinfeksi virus Covid-19 (Sudiarsa & Wiraditya, 2020).

Namun, selain manfaat dari aplikasi PeduliLindungi, penggunaan aplikasi ini juga menimbulkan masalah privasi. Contohnya pengguna aplikasi PeduliLindungi harus mengisi informasi pribadi seperti nama, alamat, NIK dan nomor ponsel untuk mendaftar akun. Setelah itu, PeduliLindungi akan meminta pengguna untuk mengaktifkan bluetooth agar bisa merekam informasi keberadaan pengguna. Jika ada perangkat lain yang terdaftar dengan PeduliLindungi di area bluetooth, terjadi pertukaran ID anonim, yang direkam di masing-masing perangkat (Olivia et al., 2020).Berdasarkan beberapa manfaat dan bahkan sampai permasalahan yang ada pada aplikasi PeduliLindungi, banyak pengguna yang memberi ulasan mengenai penggunaan aplikasi ini di google playstore dengan berbagai sudut pandang dan penilaian. Penelitian ini akan melakukan klasifikasi sentimen terhadap ulasan pengguna Aplikasi PeduliLindungi dengan menggunakan metode *K-Nearest Neighbor*. Metode ini juga sudah pernah digunakan pada beberapa penelitian dan jurnal sebelumnya. Algoritma KNN merupakan salah satu metode yang digunakan untuk analisis klasifikasi, algoritma KNN memprediksi dengan mencari jarak terpendek antara data yang dievaluasi dan *K-nearest neighbor* pada data latih (Bode, 2017).

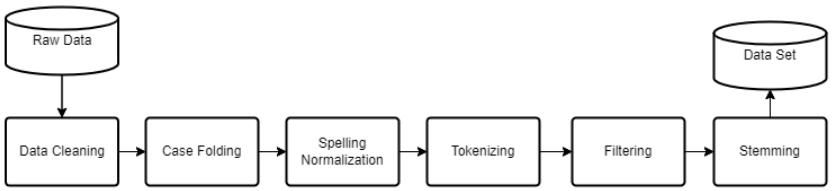
2 Tinjauan Pustaka

2.1 PeduliLindungi

PeduliLindungi adalah aplikasi yang dirancang untuk membantu instansi pemerintah terkait melacak epidemi penyakit coronavirus (Covid-19). Aplikasi ini mengandalkan partisipasi masyarakat untuk saling berbagi data lokasi saat bepergian dan melacak riwayat kontak dengan pengidap Covid-19 (Sudiarsa & Wiraditya, 2020). Aplikasi ini dirilis pada tanggal 28 Maret 2020 oleh Kementerian Komunikasi dan informatika, Kementerian Kesehatan, Kementerian BUMN dan Badan Nasional Penanggulangan bencana, yang dapat dilihat pada halaman resmi aplikasi PeduliLindungi di google play store. Aplikasi Peduli Linden dikembangkan oleh PT. Hak cipta Telekomunikasi Indonesia, Tbk, dan aplikasinya dilisensikan kepada Pemerintah Indonesia. Kementerian Komunikasi dan Informatika dan Kementerian Badan Usaha Milik Negara. 5 Aplikasi ini sangat membantu pemerintah untuk mempelajari gerakan masyarakat yang membantu memetakan masyarakat yang terpapar virus Covid19 (Olivia et al., 2020).

2.2 Praproses Data

Preprocessing adalah teknik penambangan data yang mengubah data mentah menjadi format yang terstruktur dan mudah dipahami. Dalam banyak kasus, data mentah tidak lengkap dan tidak konsisten dan dapat mengandung banyak kesalahan (Firmansyah et al., 2016). Biasanya berbagai masalah dengan data dapat mempengaruhi hasil dari proses penambangan itu sendiri, seperti nilai yang hilang, data yang berlebihan, outlier, dan format data yang tidak sesuai untuk sistem. Oleh karena itu, diperlukan langkah pretreatment untuk mengatasi permasalahan tersebut, langkah-langkah pada *text preprocessing* adalah sebagai berikut pada Gambar 1 :



Gambar 1. Tahap Praproses Teks

2.3 *Term Frequency-Inverse Document Frequency* (TF-IDF)

TF-IDF memperhitungkan kurangnya terminologi di seluruh dokumen dan melakukan pembobotan istilah untuk setiap dokumen. Inti dari TF-IDF adalah bobot konsep dari setiap dokumen (Firmansyah et al., 2016). Pengukuran ini sering digunakan sebagai faktor pembobotan untuk pencarian informasi, penambangan teks, dan pemodelan pengguna. Nilai TF-IDF meningkat sebanding dengan frekuensi kemunculan istilah dan tergantung pada jumlah dokumen dalam korpus yang berisi istilah ini. *Term Frequency* (TF) adalah pembobotan setiap kata (*term*) dalam suatu dokumen berdasarkan jumlah kemunculan dalam dokumen tersebut. Semakin sering sebuah kata muncul dalam sebuah dokumen, semakin tinggi bobot yang ditentukan (TF tinggi), sehingga frekuensi/frekuensi maksimumnya adalah istilah/kata umum biasanya merupakan topik yang ingin dicari, tetapi karena frekuensi dokumen terbalik, kata umum menjadi kata umum. *Inverse Document Frequency* (IDF) dimaksudkan untuk menentukan apakah istilah yang Anda cari cocok dengan kata kunci yang Anda cari. Istilah umum memiliki sedikit pengaruh dalam menentukan hubungan antara kata kunci dalam dokumen. Rumus TF = jumlah frekuensi kata terpilih / jumlah kata, sedangkan rumus IDF = log (jumlah dokumen /jumlah frekuensi kata terpilih). Berikut cara perhitungan TFIDF pada Persamaan (1).

$$W_{dt}(t, d.D) = TF(t, d) \times IDF(t, D) \text{ ……..... (1)}$$

Keterangan:

N : Total dokumen

df : Banyak dokumen dari kata yang dicari.

$TF(t, d)$: Jumlah kata yang dicari dalam dokumen

$IDF(t, D)$: $Log\left(\frac{N}{df}\right)$

$W_{dt}(t, d. D)$: Nilai dokumen ke-*d* pada kata ke-*t*

2.4 Klasifikasi KNN

K-Nearest Neighbor (KNN) adalah algoritma yang dapat mengklasifikasikan objek berdasarkan data pelatihan terkait objek. Data pelatihan yang digunakan dianggap sebagai ruang multidimensi yang dimensinya mewakili properti dari setiap data. Metode JST sangat sederhana dalam sistem kerjanya, cukup memproyeksikan data latih ke dalam ruang multidimensi untuk menentukan JST, dan biasanya dihitung berdasarkan jarak Euclidean (Kurniawan et al., 2020). Dalam penggunaan algoritma *k-nearest neighbor*, perlu menentukan jumlah k-tetangga terdekat yang dipakai untuk mengklasifikasikan data baru. Banyaknya k, sebaiknya ganjil, misalnya k = 1, 2, 3, dan seterusnya. KNN memiliki keunggulan dibandingkan dataset pelatihan dengan banyak *noise* dan efisien dengan jumlah data pelatihan yang tinggi/besar. Namun kekurangan dari KNN adalah masih diperlukannya penentuan nilai K dan untuk pemilihan atribut yang terbaik (Bode, 2017). Berikut adalah rumus menghitung jarak antar dokumen uji da dokumen latih pada Persamaan (2).

$$Euclidean\ Distance\ (x, y) = \sqrt{\sum_{i=1}^{t} (x_i - y_i)^2} \ldots\ldots\ldots (2)$$

Keterangan :

t : Jumlah kata

x_i : Dokumen uji

y_i : Dokumen latih

Penerapan metode *K-Nearest Neighbor*. Penerapan metode K-NN melalui beberapa tahapan:

1. Menentukan parameter k
2. Hitung jarak antara data yang dievaluasi dan semua pelatihan
3. Urutkan jarak formasi (dari paling kecil ke paling besar nilainya)
4. Tentukan jarak terdekat orde k
5. Cocokkan kelas yang sesuai
6. Temukan jumlah kelas dari tetangga terdekat dan atur kelas tersebut sebagai lapisan data untuk evaluasi (Suwirmayanti, 2017).

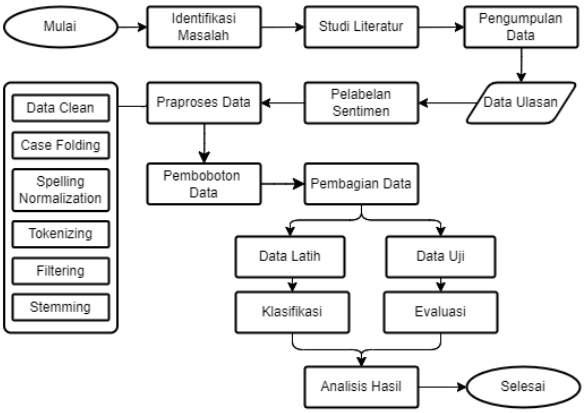
2.5 SMOTE (*Synthetic Minority Oversampling Technique*)

Metode SMOTE (*Synthetic Minority Oversampling Technique*) adalah metode populer yang digunakan untuk menangani ketidakseimbangan kelas, teknik ini mensintesis sampel baru dari kelas minoritas untuk menyeimbangkan kumpulan data dengan melakukan *resampling* kelas minoritas (Siringoringo, 2018). Langkah-langkah proses SMOTE :

1. Ambil sampel random contohnya P1
2. Kemudian terapkan algoritma KNN pada P1
3. Ambil jarak tetangga terdekat P1, contohnya P2
4. Lalu *generate* data baru P1’ = P1 + rand(0,1)*(P2-P1)
5. Ulang terus proses tersebut hingga jumlah data minoritas sebanyak data mayoritas

3 Metodologi Penelitian

Tahap yang dilakukan untuk mencapai tujuan penelitian yang telah dirumuskan dan dapat dilihat pada Gambar 2 proses dari penelitian ini :



Ganmbar 2. Alur Penelitian

Identifikasi Masalah. Proses identifikasi dari penelitian ini mengangkat permasalahan yang ada di aplikasi PeduliLindungi dengan mendapatkan informasi berupa hasil analisa sentimen positif dan sentimen negatifnya. Terutama dalam kondisi pandemi aplikasi ini sangat banyak digunakan hampir seluruh masyarakat di Indonesia, namun dengan berbagai komentar, saran, kritik dan juga kontroversi yang ada maka penelitian ini dilakukan supaya kiranya menjadi informasi yang berguna bagi masyarakat maupun pihak yang berkepentingan terhadap aplikasi ini.

Studi literatur. Tujuan pada tahap ini yaitu sebagai pendukung dalam memecahkan masalah dalam penelitian yang digunakan sebagai sumber pustaka dengan cara mengumpulkan jurnal dan buku yang relevan mengenai *text mining* atau klasifikasi sentimen dan beberapa penggunaan algoritma KNN.

Scraping Data. Data yang berhasil diperoleh adalah ulasan pengguna pada tanggal 07 September 2021 sebanyak 1.000 data ulasan, hal ini dikarenakan pada tanggal tersebut pemerintah mewajibkan menggunakan aplkasi PeduliLindungi yang diatur dalam Inmendagri Nomor 38 Tahun 2021.

Pelabelan kelas sentimen. Data teks yang telah melewati praproses akan ditandai dengan dua kelas yaitu positif dan negatif. Pelabelan dilakukan secara manual yang akan dilakukan oleh tiga *user* yang berbeda dan akan memberikan label menurut hasil pendapat ketiga *user* tersebut.

Praproses. Langkah selanjutnya adalah melakukan praproses data dokumen dalam format teks untuk membersihkan kalimat dan karakter yang tidak perlu untuk mendapatkan data yang lebih terstruktur. Tahapan *preprocessing* yang harus dilakukan adalah

1. *Data cleaning*, proses ini menghapus karakter yang tidak pantas seperti tanda baca, angka, URL, huruf besar, spasi, tagar, karakter tunggal, dan banyak lagi.
2. *Case folding*, proses mengubah bentuk sebuah kata menjadi huruf kecil atau huruf besar (lowercase).
3. *Spelling normalization*, yaitu langkah pemilihan kata tidak baku, salah ejaan, dan singkatan untuk dikoreksi menjadi kata yang sesuai dengan kamus KBBI.
4. *Tokenizing* adalah proses pemecahan kalimat dalam dokumen menjadi kata-kata terpisah (token).
5. *Filtering* yaitu proses penghilangan kata-kata yang dianggap kurang penting dari hasil tokenisasi menggunakan kamus kata-kata terlarang bahasa Indonesia.
6. *Stemming* adalah fase stemming juga merupakan langkah yang diperlukan untuk mengurangi jumlah berbagai subscript dari datum untuk mengembalikan kata-kata dengan akhiran atau awalan ke bentuk dasarnya.

Pembobotan Kata. Pada tahapan pembobotan membantu dalam menetapkan bobot kata ke data dokumen. Nilai-nilai tersebut akan digunakan di tahap klasifikasi. Pembobotan dilakukan dengan menggunakan pembobotan TF-IDF dengan memberikan nilai pada data ulasannya. Hasil akhir bobot ini merupakan hasil perkalian dari TF dan IDF dengan menggunakan persamaan (1).

Pembagian Data. Dalam penelitian ini, pembagian data dilakukan dengan membaginya menjadi dua bagian yaitu data latih dan data uji, dengan perbandingan yang digunakan adalah data latih sebesar 80% dan data uji

sebesar 20%.

Klasifikasi. Proses klasifikasi merupakan bagian untuk mendapatkan informasi untuk membantu dalam membuat keputusan. Metode *machine learning* yang digunakan untuk mengklasifikasikan data teks dalam penelitian ini adalah metode *K-Nearest Neighbor* yang menggunakan perbandingan data 80:20 dengan data latih 80% dan data uji 20% untuk memverifikasi keakuratan pemodelan klasifikasi. Nantinya di proses Klasifikasi ini akan dilakukan dengan tiga data berbeda jumlah, yang pertama merupakan dengan menggunakan data yang tidak seimbang, dan yang kedua menggunakan data yang sudah seimbang dengan cara *undersampling,* dan yang ketiga menggunakan data yang sudah seimbang dengan teknik *oversampling* SMOTE.

Evaluasi. Evaluasi yang akan dilakukan untuk mengukur ketepatan klasifikasi menggunakan metode *confusion matrix* dengan nilai akurasi, presisi, sensitivitas/*recall,* spesifisitas dan F1 Score. Nantinya data training digunakan untuk tahap klasifikasi, sedangkan data testing digunakan untuk tahap evaluasi.

4 Hasil dan Pembahasan

Data yang digunakan adalah ulasan pada tanggal 07 September 2021 sebanyak 1000 data ulasan. Proses pengambilan data ulasan pada google play pada penelitian ini menggunakan teknik *web scraping* dengan *library google play scraper* yang tersedia pada python, dapat dilihat sampel hasil dari pengambilan data pada Tabel 1.

Tabel 1. Sampel Hasil Pengambilan Data

userName	score	at	content
H Oio	3	2021-09-07 0:00:35	Opsi perbaharui status sangat merepotkan, krn kl lagi di mall atau ruang publik lainnya hrs update dan memakan wkt. Apakah aplikasi tdk bs mengupdate data secara otomatis?
adams. yoo	5	2021-09-07 0:01:10	Lebih baik dari aplikasi sebelumnya, hanya kadang masih lambat menunjukkan titik lokasi di map nya. Selebihnya wokeh.
Noes Noes Ajah	5	2021-09-07 0:02:03	Memberikan dan menjelaskan identitas serta hasil pemeriksaan secara akurat dan jelas
Rizky Adhi	2	2021-09-07 0:02:57	Keseringan pembaharuan app

4.1 Pelabelan Data

Setelah melakukan pengambilan data dan sudah diperoleh 1.000 data ulasan dari aplikasi PeduliLindungi selanjutnya data ulasan ini akan dilabeli. Cara pelabelan yang digunakan pada penelitian ini yaitu dilakukan secara manual oleh tiga annotator. Data ulasan yang sudah dilabeli oleh masing-masing annotator akan menentukan label dari setiap ulasan dengan total terbanyak label yang dipilih. Berikut adalah hasil anotasi dari tiga annotator dapat dilihat pada Tabel 2.

Tabel 2. Hasil Pelabelan Data

Label	Annotator 1	Annotator 2	Annotator 3	3 label sama	2 label sama	Hasil Label
Positif	238	273	251	230	20	250
Negatif	762	727	749	718	32	750

4.1 Praproses Data

Data ulasan yang ditarik mengandung simbol dan kata-kata yang kurang deskriptif, sehingga peru dilakukan

praproses data untuk mengklasifikasikan data tersebut.

1. ***Data Cleaning,*** pada proses ini data ulasan akan dibersihkan dengan membuang karakter yang tidak digunakan seperti URL, username, hashtag, simbol, tanda baca, dll. Contoh nya data ulasan "Aplikasi berguna, cuman sering update", setelah dilakukan *data cleaning* menjadi "Aplikasi berguna cuman sering update"
2. ***Case folding*** dilakukan dengan bantuan fungsi *lower* () yang mengubah semua teks menjadi huruf kecil. Contohnya, dari *data cleaning* "Aplikasi berguna cuman sering update" menjadi data data *case folding* "aplikasi berguna cuman sering update".
3. ***Spelling normalization*** yaitu proses dimana data tersebut akan diperbaiki ejaannya agar sesuai dengan Kamus Besar Bahasa Indonesia (KBBI) dengan daftar kata *normalization.* Contohnya, data *case folding* "aplikasi berguna cuman sering update" menjadi "aplikasi berguna cuman sering perbarui"
4. ***Tokenizing*** dilakukan untuk memisahkan setiap kata pada kalimat ulasan dengan spasi (*whitespace*) sebagai pemisah kata. Contohnya, data *Spelling normalization* "aplikasi berguna cuman sering perbarui" menjadi ['aplikasi', 'berguna', 'cuman', 'sering', 'perbarui']
5. ***Filtering*** atau sering disebut juga dengan *stopword removal* yaitu menghapus kata yang dianggap kurang penting seperti tidak memiliki makna yang penting dan tidak penting untuk proses klasifikasi. Contohnya, data *tokenizing* ['aplikasi', 'berguna', 'cuman', 'sering', 'perbarui'] menjadi "sertifikat vaksin muncul aplikasi berfungsi"
6. ***Stemming*** dilakukan untuk memeriksa kata yang memiliki imbuhan dan akan diubah menjadi kata dasar dengan bantuan *library* sastrawi. Contohnya, data *filtering* "sertifikat vaksin muncul aplikasi berfungsi" menjadi "aplikasi guna cuman baru"

4.2 Pembobotan data dengan TF IDF

Melalui praproses didapatkan 1275 kata dan harus diubah menjadi angka untuk bisa dikerjakan oleh mesin, hal ini yang menyebabkan perlunya menggunakan pembobotan TFIDF, nilai TFIDF yang sudah didapatkan bisa dilihat hasil sampel dari perhitungan TFIDF juga akan ditampilkan pada Tabel 3.

Tabel 3. Nilai TFIDF

Term	Dokumen			DF	IDF	TF-IDF		
	D1	D2	D3			D1	D2	D3
sertifikat	0.2	0	0	1	0.477	0.095	0	0
vaksin	0.2	0	0	1	0.477	0.095	0	0
muncul	0.2	0	0	1	0.477	0.095	0	0
aplikasi	0.2	0.25	0	2	0.176	0.035	0.044	0
fungsi	0.2	0	0	1	0.477	0.095	0	0
guna	0	0.25	0	1	0.477	0	0.119	0
cuman	0	0.25	0	1	0.477	0	0.119	0
baru	0	0.25	0	1	0.477	0	0.119	0
masuk	0	0	0.5	1	0.477	0	0	0.239
akun	0	0	0.25	1	0.477	0	0	0.119
payah	0	0	0.25	1	0.477	0	0	0.119

4.3 *Word Cloud* data fitur

Setelah melalui praproses dapat dilihat daftar kata yang paling sering muncul pada label positif. Ada tiga

kata yang paling sering muncul yaitu ‘bantu’, ‘aplikasi’, dan ‘covid’. Berikut adalah gambar dari *word cloud* label positif pada Gambar 3.

Gambar 3. *Word Cloud* Label Positif

Setelah melalui praproses dapat dilihat daftar kata yang paling sering muncul pada label negatif. Ada tiga kata yang paling sering muncul yaitu ‘aplikasi’, ‘masuk’, dan ‘baru’. Berikut adalah gambar dari *word cloud* label negatif pada Gambar 4.

Gambar 4. *Word Cloud* Label Negatif

4.4 SMOTE (*Synthetic Minority Oversampling Technique*)

Dalam penelitian ini data yang digunakan tidak seimbang dengan data berlabel positif sebanyak 250 dan data berlabel negatif sebanyak 750. Contoh terdapat kata ‘sertifikat’, ‘vaksin’, ‘muncul’, ‘aplikasi’, ‘fungsi’. Berarti terdapat 5 fitur sebagai berikut pada Tabel 4

Tabel 4. Sampel Data untuk SMOTE

	sertifikat	**vaksin**	**muncul**	**aplikasi**	**fungsi**
P1	0.2	0.2	0.2	0.2	0.2
P2	0	0	0	0.25	0

Berikut menghitung jarak tetangga dengan rumus *Euclidean Distance*

$$d = \sqrt{(0.2 - 0)^2 + (0.2 - 0)^2 + (0.2 - 0)^2 + (0.2 - 0.25)^2 + (0.2 - 0)^2}$$

$$d = 0.4031128874$$

P1 disini sebagai sampel untuk dapat disintesiskan dan P2 adalah salah satu tetangganya yang terdekat dari 5 tetangga lainnya dengan jarak 0.4031128874. Berikut adalah perhitungan P1’ untuk membuat data sintetis dengan menggunakan metode SMOTE.

P1’ = P1 + rand(0,1) * (P2 – P1)

P1(x1,x2,x3,x4,x5) = (0.2, 0.2, 0.2, 0.2, 0.2)
P2(x1,x2,x3,x4,x5) = (0, 0, 0, 0.25, 0)
P2 (x1) – P1 (x1) = 0 - 0.2 = - 0.2
P2 (x2) – P1 (x2) = 0 - 0.2 = - 0.2
P2 (x3) – P1 (x3) = 0 - 0.2 = - 0.2
P2 (x4) – P1 (x4) = 0. - 0.25 = - 0.25
P2 (x5) – P1 (x5) = 0 - 0.2 = - 0.2
P1'(x1',x2',x3',x4',x5') = (0.2, 0.2, 0.2, 0.2, 0.2) + rand(0,1) * (-0.2, -0.2, -0.2, -0.25, -0.2)
Setelah mendapatkan selisih P2 dengan P1, maka bisa dimasukan nilai rand, misalnya rand(0,1) = 0.1.
P1'(x1',x2',x3',x4',x5') = (0.2, 0.2, 0.2, 0.2, 0.2) + 0,1 * (-0.2, -0.2, -0.2, -0.25, -0.2)
P1'(x1',x2',x3',x4',x5') = (0.2, 0.2, 0.2, 0.2, 0.2) + (-0.02, -0.02, -0.02, -0.025, -0.2)
P1'(x1',x2',x3',x4',x5') = (0.18, 0.18, 0.18, 0.175, 0.18)
Setelah melakukan perhitungan didapatkan data sintesis baru dari perhitungan menggunakan teknik SMOTE. Pada Tabel 5 ini dapat terlihat data baru yang dibentuk dari hasil *oversampling* menggunakan SMOTE.

Tabel 5. Hasil Setelah *Oversampling* SMOTE

	sertifikat	**vaksin**	**muncul**	**aplikasi**	**fungsi**
P1	0.2	0.2	0.2	0.2	0.2
P2	0	0	0	0.25	0
P1'	0.18	0.18	0.18	0.175	0.18

4.3 Klasifikasi

Model yang digunakan pada penelitian ini adalah metode *K-Nearest Neighbor*, dimana proses klasifikasi ini nantinya akan menggunakan nilai setiap fitur dari hasil TF-IDF sebelumnya dan untuk menghitung jarak *Euclidean Distance* menggunakan *library* KNeighborsClassifier. Jarak *Euclidean Distance* yang dihitung yaitu antara data yang akan diuji dengan dengan data latih dengan menggunakan Persamaan (2). Setelah itu nantinya nilai tersebut akan diurutkan dari paling kecil sampai nilai ke paling besar, nilai yang paling kecil menunjukan bahwa data yang diuji semakin menyerupai data latih tersebut dan nilai *Euclidean Distance* yang paling kecil adalah 0. Model pada penelitian ini menggunakan split 80:20 yaitu dengan data latih sebanyak 80% dari data keseluruhan, yang dimana sudah memiliki label positif dan negatif. Sedangkan untuk data uji sebanyak 20% dari data keseluruhan yang belum mendapatkan label.

4.4 Evaluasi

Sesudah melalui proses klasifikasi tersebut maka dilakukan evaluasi untuk mengetahui bagaimana performa model yang telah dilakukan, dengan membandingkan antara label sentimen yang diprediksi dengan label sentimen yang sebenarnya pada data uji. Dalam penelitian ini terdapat tiga data yang berbeda yang nantinya hasil dari evaluasi tersebut akan dibandingkan, ketiga data tersebut yaitu yang pertama menggunakan data yang tidak seimbang, yang kedua menggunakan data yang sudah di *undersampling*, lalu yang ketiga menggunakan data yang sudah di *oversampling* menggunakan metode SMOTE. Ketiga penelitian ini diuji dengan beberapa jumlah tetangga yaitu 1, 2, 3, 4, 5, 6, 7, 8, 9, dan 10. Berikut adalah Akurasi presisi, F1 score, spesifisitas, dan sensitivitas dari pengujian terhadap data yang tidak seimbang.

1. Berikut adalah performa dari klasifikasi dengan data tidak seimbang dapat dilihat pada Tabel 6.

Tabel 6. Hasil evaluasi dengan data tidak seimbang

K	**Akurasi**	**Precision**	**F1 Score**	**Spesifisitas**	**Sensitivitas**
1	0.83	0.8571	0.5142	0.9801	0.3673
2	0.83	1	0.4687	1	0.3061
3	0.84	1	0.5151	1	0.3469
4	0.825	1	0.4444	1	0.2857
5	0.875	0.9285	0.6753	0.9867	0.5306

6	0.865	0.9583	0.6301	0.9933	0.4693
7	0.915	0.9444	0.7999	0.9867	0.6938
8	0.895	0.9666	0.7341	0.9933	0.5918
9	0.905	0.9166	0.7764	0.9801	0.6734
10	0.9	0.9393	0.756	0.9867	0.6326
Rata - rata	**0.868**	**0.95108**	**0.63142**	**0.99069**	**0.48975**

Dalam klasifikasi data tidak seimbang ini juga akurasi tertinggi terdapat pada K = 7 dengan nilai akurasi sebesar 0.915, nilai presisi sebesar 0.9444, nilai F1 score 0.7999, nilai spesifisitas sebesar 0.9867, dan nilai sensitivitas sebesar 0.6938. Selisih dari nilai spesifisitas dan nilai sensitivitas sebesar 0,2929.

2. Klasifikasi dengan *Undersampling*

Pada klasifikasi kedua yang digunakan adalah data yang diseimbangkan dengan cara *undersampling*, yaitu melakukan pengurangan data yang mayoritas menjadi sama banyak dengan data minoritas, sehingga pada data *undersampling* ini menjadikan data positif sebanyak 250 dan data negatif sebanyak 250. Berikut hasilnya pada Tabel 7.

Tabel 7. Hasil evaluasi dengan *undersampling*

K	**Akurasi**	**Precision**	**F1 Score**	**Spesifisitas**	**Sensitivitas**
1	0.64	0.909	0.5263	0.9565	0.3703
2	0.61	0.9411	0.4507	0.9782	0.2962
3	0.82	0.875	0.8235	0.8695	0.7777
4	0.85	0.8823	0.8571	0.8695	0.8333
5	0.86	0.9761	0.8541	0.9782	0.7592
6	0.81	0.8571	0.8155	0.8478	0.7777
7	0.88	0.9772	0.8775	0.9782	0.7962
8	0.89	0.9387	0.8932	0.9347	0.8518
9	0.84	0.8958	0.8431	0.8913	0.7962
10	0.87	0.9767	0.8659	0.9782	0.7777
Rata - rata	**0.807**	**0.9229**	**0.78069**	**0.92821**	**0.70363**

Dalam klasifikasi ini juga akurasi tertinggi terdapat pada K = 8 dengan nilai akurasi sebesar 0.89, nilai presisi sebesar 0.9387, nilai F1 score 0.8932, nilai spesifisitas sebesar 0.9347, dan nilai sensitivitas sebesar 0.8518. Selisih dari nilai spesifisitas dan nilai sensitivitas sebesar 0,0829 dimana angka ini juga menunjukan bahwa selisihnya lebih rendah daripada klasifikasi data tidak seimbang.

3. Klasifikasi dengan *Oversampling*

Pada klasifikasi ini yang digunakan adalah data yang diseimbangkan dengan cara *oversampling*, sehingga data positif sebanyak 750 dan data negatif sebanyak 750 yang diuji. Berikut adalaha hasilnya pada Tabel 8.

Tabel 8. Hasil Evaluasi dengan *Oversampling*

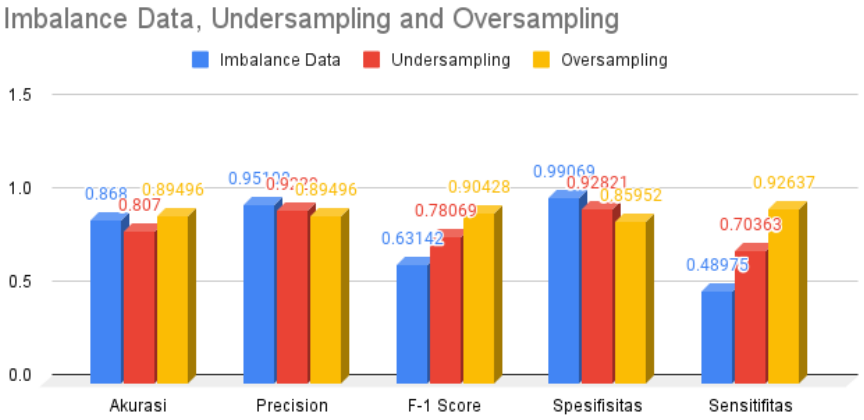
K	**Akurasi**	**Precision**	**F1 Score**	**Spesifisitas**	**Sensitivitas**
1	0.9766	0.9691	0.9781	0.9645	0.9874

2	0.9466	0.9798	0.948	0.9787	0.9182
3	0.93	0.948	0.9329	0.9432	0.9182
4	0.8933	0.9503	0.8933	0.9503	0.8427
5	0.89	0.8579	0.9014	0.8226	0.9496
6	0.88	0.8773	0.8881	0.8581	0.8993
7	0.8566	0.8186	0.8739	0.7659	0.9371
8	0.8733	0.8457	0.8862	0.8085	0.9308
9	0.8466	0.8021	0.867	0.7375	0.9433
10	0.8566	0.8186	0.8739	0.7659	0.9371
Rata - rata	**0.89496**	**0.88674**	**0.90428**	**0.85952**	**0.92637**

Dalam klasifikasi dengan *undersampling* ini juga akurasi tertinggi terdapat pada K = 1 dengan nilai akurasi sebesar 0.9766, nilai presisi sebesar 0.9691, nilai F1 score 0.9781, nilai spesifisitas sebesar 0.9645, dan nilai sensitivitas sebesar 0.9874. Selisih dari nilai spesifisitas dan nilai sensitivitas sebesar 0,0229, dari akurasi tertinggi setiap klasifikasi hasil ini adalah selisih terendah yang berarti bahwa akurasi ini mengklasifikasikan data negatif dan data positif dengan cukup baik.

4.1.1. Perbandingan rata-rata

Dari ketiga klasifikasi tersebut dapat diamati rata-rata pada masing-masing klasifikasi yang dilakukan. Berikut Gambar 5 adalah gambar dari grafik perbandingan rata-rata performa dari ketiga klasifikasi.



Gambar 5. Grafik perbandingan rata-rata

Pada Gambar 5 dapat dilihat pada grafik batang dari data yang tidak seimbang, data yang di *undersampling* dan data yang di *oversampling*. Terlihat pada data yang sudah di *oversampling* mendapatkan rata-rata performa yang lebih stabil, sedangkan untuk data yang sudah di *undersampling* memiliki performa yang kurang stabil dibandingkan performa dengan menggunakan data yang sudah di *oversampling*, dan pada data yang tidak seimbang memiliki rata-rata performa yang sangat tidak stabil dan perbedaan spesifisitas dengan sensitivitas sangat jauh, sehingga dapat disimpulkan bahwa performa pada data yang seimbang dengan SMOTE adalah yang paling baik.

5 Kesimpulan

Klasifikasi sentimen ulasan pengguna aplikasi PeduliLindungi dengan data sebanyak 1000 yang diberi dua label yaitu label positif dan negatif dengan jumlah label positif 250 dan label negatif 750. Data yang sudah dilabeli akan dilakukan praproses data dengan cara *data cleaning, case folding, spelling normalization, tokenizing, filtering - stopword removal,* dan *stemming.* Kemudian data tersebut akan

dilakukan pembobotan dengan *Term Frequency – Inverse Document Frequency* (TF - IDF) supaya dapat diproses oleh model *machine learning.* Setelah itu akan dilakukan tiga klasifikasi dengan banyak data yang berbeda, namun sebelum dilakukan klasifikasi data akan di bagi menjadi data latih sebesar 80% dan data uji sebesar 20%. Ketiga jenis data tersebut antara lain yang pertama dengan menggunakan data yang tdak seimbang, lalu yang kedua menggunakan data yang seimbang dengan metode *undersampling* dan yang ketiga menggunakan data yang seimbang dengan metode *oversampling* SMOTE.
Dari ketiga hasil tersebut dapat disimpulkan

1. Sentimen ulasan pengguna aplikasi PeduliLindungi, lebih banyak sentimen negatif daripada sentimen positif yang mana terdapat 750 negatif dan 250 positif.
2. Performa klasifikasi sentimen ulasan pengguna aplikasi PeduliLindungi yang dilakukan menggunakan algoritma *K-Nearest Neighbor* memiliki performa yang paling baik ada pada penelitian yang menggunakan teknik SMOTE, pada K = 1 dengan nilai akurasi sebesar 0.9766, nilai presisi sebesar 0.9691, nilai F1 score 0.9781, nilai spesifisitas sebesar 0.9645, dan nilai sensitivitas sebesar 0.9874.
3. Perbandingan performa dari ketiga data penelitian tersebut mendapatkan hasil rata-rata yang terus meningkat pada data yang seimbang dengan teknik *undersampling* dan hasil rata-rata terbaik pada teknik SMOTE. Ini menunjukan bahwa performa klasifikasi sangat baik menggunakan data yang seimbang.

6 Daftar Pustaka

[1] Firmansyah, R. F. N., Fauzi, M. A., & Afirianto, T. (2016). Sentiment Analysis pada Review Aplikasi Mobile Menggunakan Metode Naive Bayes dan Query Expansion. *Doro Ptiik*, *8*(December), 14.

[2] Fridom Mailo, F., Lazuardi, L., Manajemen dan kebijakan Kesehatan Fakultas Kedokteran, D., Masyarakat dan Keperawatan Universitas Gadjah Mada, K., Sistem Informasi Manajemen Kesehatan Fakultas Kedokteran, D., Masyarakat dan Keperawatan, K., & Gadjah Mada, U. (2019). Analisis Sentimen Data Twitter Menggunakan Metode Text Mining Tentang Masalah Obesitas di Indonesia. *Jurnal Sistem Informasi Kesehatan Masyarakat Journal of Information Systems for Public Health*, *4*(1), 28–36. https://jurnal.ugm.ac.id/jisph/article/view/44455

[3] Kurniawan, M. A. A., Ermatita, E., & Falih, N. (2020). Pemanfaatan Pengolahan Citra dan Klasifikasi K-Nearest Neighbor pada Citra Telur Ayam. *Informatik : Jurnal Ilmu Komputer*, *16*(3), 164. https://doi.org/10.52958/iftk.v16i3.2131

[4] Roihan, A., Sunarya, P. A., & Rafika, A. S. (2020). Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper. *IJCIT (Indonesian Journal on Computer and Information Technology)*, *5*(1), 75–82. https://doi.org/10.31294/ijcit.v5i1.7951

[5] Sari, F. V., & Wibowo, A. (2019). Analisis Sentimen Pelanggan Toko Online Jd. Id Menggunakan Metode Naïve Bayes Classifier Berbasis Konversi Ikon Emosi. *Simetris: Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer*, *2*(2), 681–686.

[6] Adhi Putra, A. D., & Juanita, S. (2021). Analisis Sentimen pada Ulasan pengguna Aplikasi Bibit Dan Bareksa dengan Algoritma KNN. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, *8*(2), 636–646. https://doi.org/10.35957/jatisi.v8i2.962

[7] Fitrianti, R. P., Kurniawati, A., & Agusten, D. (2019). Implementasi Algoritma K - Nearest Neighbor Terhadap Analisis Sentimen Review Restoran Dengan Teks Bahasa Indonesia. *Journal of Chemical Information and Modeling*, *53*(9), 1689–1699.

[8] Isnain, A. R., Supriyanto, J., & Kharisma, M. P. (2021). Implementation of K-Nearest Neighbor (K-NN) Algorithm For Public Sentiment Analysis of Online Learning. *Teknik Informatika Dan Sistem Informasi*, *15*(2), 121–130. https://doi.org/10.22146/ijccs.65176

[9] Nurhidayati, N., Sugiyah, S., & Yuliantari, K. (2021). Pengaturan Perlindungan Data Pribadi Dalam Penggunaan Aplikasi Pedulilindungi. *Widya Cipta: Jurnal Sekretari Dan Manajemen*, *5*(1), 39–45. https://doi.org/10.31294/widyacipta.v5i1.9447

[10] Olivia, D. O., Rosadi, S. D., & Permata, R. R. (2020). PERLINDUNGAN DATA PRIBADI DALAM PENYELENGGARAAN APLIKASI SURVEILANS KESEHATAN PEDULILINDUNGI DAN COVIDSAFE DI INDONESIA DAN AUSTRALIA. *Sustainability (Switzerland)*, *4*(1), 1–9. https://pesquisa.bvsalud.org/portal/resource/en/mdl-20203177951%0Ahttp://dx.doi.org/10.1038/s41562-020-0887-

9%0Ahttp://dx.doi.org/10.1038/s41562-020-0884-z%0Ahttps://doi.org/10.1080/13669877.2020.1758193%0Ahttp://sersc.org/journals/index.php/IJAST/article

[11] PDPI. (2020). A comparison of the Indian Health Service counseling technique with traditional, lecture-style counseling. In *Journal of the American Pharmacists Association* (Vol. 55, Issue 5). https://doi.org/10.1331/JAPhA.2015.14093

[12] Sari, R. (2020). Analisis Sentimen Pada Review Objek Wisata Dunia Fantasi Menggunakan Algoritma K-Nearest Neighbor (K-Nn). *EVOLUSI : Jurnal Sains Dan Manajemen*, *8*(1), 10–17. https://doi.org/10.31294/evolusi.v8i1.7371

[13] Sudiarsa, I. W., & Wiraditya, I. G. B. (2020). Analisis Usability Pada Aplikasi Peduli Lindungi Sebagai Aplikasi Informasi Dan Tracking Covid-19 Dengan Heuristic Evaluation. *Journal of Information Technology and Computer Sains*, *3*(2), 354–364.

[14] Susilo, A., Rumende, C. M., Pitoyo, C. W., Santoso, W. D., Yulianti, M., Herikurniawan, H., Sinto, R., Singh, G., Nainggolan, L., Nelwan, E. J., Chen, L. K., Widhani, A., Wijaya, E., Wicaksana, B., Maksum, M., Annisa, F., Jasirwan, C. O. M., & Yunihastuti, E. (2020). Coronavirus Disease 2019:

[15] Bode, A. (2017). K-Nearest Neighbor dengan Feature Selection Menggunakan Backward Elimination untuk Prediksi Harga Komoditi Kopi Arabika. *ILKOM Jurnal Ilmiah*, *9*(2), 188–195.

[16] Tuhuteru, H. (2020). Analisis Sentimen Masyarakat Terhadap Pembatasan Sosial Berskala Besar Menggunakan Algoritma Support Vector Machine. *INFORMATION SYSTEM DEVELOPMENT (ISD)*, 5(2), 7-13.

[17] Siringoringo, R. (2018). Klasifikasi Data Tidak Seimbang Menggunakan Algoritma SMOTE dan *K-Nearest Neighbor*. Jurnal ISD, 3(1), 44-49.

[18] Salbilla, W. I., & Vista. (2021). Implementasi SMOTE dan Under Sampling pada Imbalanced Dataset untuk Prediksi Kebangkrutan Perusahaan